

**AMENDMENTS TO THE CLAIMS**

1. (Previously presented) A method of forming a peer-to-peer group within a peer-to-peer cloud, the group having a plurality of peer nodes, the method comprising the steps of:
  - selecting at least one friendly name for the group (NAME<sub>G</sub>), wherein the friendly name collectively identifies the peer-to-peer group;
  - calculating a category identification (CID) for the group from the at least one friendly name; and
  - providing the CID to at least one peer node within the peer-to-peer cloud, wherein the CID allows the at least one peer node within the peer-to-peer cloud to communicate directly to the peer-to-peer group.
2. (Original) The method of claim 1, wherein the step of calculating the CID comprises the step of hashing the NAME<sub>G</sub> with a seed GROUP.
3. (Original) The method of claim 1 wherein the peer-to-peer group is to be private, wherein the step of providing the CID to a peer comprises the step of sending the CID to the peer out of band.
4. (Original) The method of claim 1 wherein the peer-to-peer group is to be public, wherein the step of providing the CID to a peer comprises the step of registering the CID with a peer-to-peer name resolution protocol (PNRP) for discovery therethrough.
5. (Original) The method of claim 4, further comprising the steps of concatenating a peer ID with the CID to derive CID:ID, and registering CID:ID with PNRP for discovery therethrough.
6. (Original) The method of claim 1, further comprising the steps of receiving a connect message from the peer and returning a welcome message to the peer.
7. (Original) The method of claim 6, further comprising the steps of calculating a signature of a group object database, and sending the signature to the peer.

8. (Original) The method of claim 7, wherein the step of calculating the signature of the group object database comprises the steps of combining a unique identification (UID), sequence number (SNUM), and age for each group object in the group object database, and sorting a list of the combined UIDs/SNUMs/ages for the group objects.

9. (Original) The method of claim 7, further comprising the steps of receiving a request for specific group objects from the peer, and transmitting the specific group objects to the peer.

10. (Original) The method of claim 1, further comprising the steps of receiving a connect message from the peer and returning a refuse message to the peer along with a list of other members of the group.

11. (Previously presented) A computer-readable storage medium storing computer-executable instructions for performing the method of claim 1.

12. (Withdrawn) A method of discovering a peer-to-peer group, comprising the steps of learning a category identification (CID) for the group, calculating at least one utility peer identification, concatenating the CID with the at least one utility peer identification, searching for an instance of the concatenated CID with the at least one utility peer identification.

13. (Withdrawn) The method of claim 12, further comprising the steps of receiving a result identifying a group member, and contacting a group member with a connect message to join the group.

14. (Withdrawn) The method of claim 13, further comprising the steps of determining if the group uses contacts, and when the group does not use contacts concatenating the CID with a local peer identification (ID<sub>p</sub>) to derive CID:ID<sub>p</sub> and registering CID:ID<sub>p</sub> with PNRP for discovery therethrough.

15. (Withdrawn) The method of claim 13, further comprising the steps of determining that the group uses contacts and that a local peer is to be a contact, and

concatenating the CID with a local peer identification ( $ID_p$ ) to derive  $CID:ID_p$  and registering  $CID:ID_p$  with PNRP for discovery therethrough.

16. (Withdrawn) The method of claim 12, wherein the group is private and wherein the step of learning a category identification (CID) for the group comprises the step of learning the CID out of band.

17. (Withdrawn) The method of claim 12, wherein the group is public and wherein the step of learning a category identification (CID) for the group comprises the step of searching for the CID through PNRP.

18. (Withdrawn) The method of claim 12, wherein the step of calculating at least one utility peer identification comprises the steps of calculating  $ID_{MIN}$ ,  $ID_{MID}$ ,  $ID_{MAX}$ ,  $ID_{HMID}$ , and  $ID_{LMID}$ .

19. (Withdrawn) The method of claim 18, further comprising the steps of receiving a result identifying a result group identification ( $CID_R$ ) that is not equal to the CID, determining if  $CID_R < CID$ , and when  $CID_R < CID$  concatenating  $CID:ID_{HMID}$  and searching for  $CID:ID_{HMID}$ , else concatenating  $CID:ID_{LMID}$  and searching for  $CID:ID_{LMID}$ .

20. (Withdrawn) The method of claim 18, further comprising the step of enumerating discoverable instances for the CID.

21. (Withdrawn) The method of claim 20, wherein the step of enumerating discoverable instances for the CID comprises the steps of placing  $[ID_{MIN}, ID_{MAX}]$  in a queue Q, creating a list L, dequeuing a first ID from a front of Q having  $ID = [ID_F, ID_L]$ ; setting  $ID_C = (ID_L - ID_F)/2$ , concatenating  $CID:ID_C$ , searching for  $CID:ID_C$ , receiving a result of  $CID_R:ID_R$ , and when  $CID_R$  is equal to CID and  $ID_R$  is not in L adding  $ID_R$  to L.

22. (Withdrawn) The method of claim 21, further comprising the steps of inserting  $[ID_F, ID_R - 1]$  in Q when  $ID_F < ID_R$ .

23. (Withdrawn) The method of claim 22, further comprising the steps of inserting  $[ID_R + 1, ID_L]$  in Q when  $ID_L < ID_R$ .

24. (Withdrawn) The method of claim 18, further comprising the step of enumerating all descendants for the CID.

25. (Withdrawn) The method of claim 24, wherein the step of enumerating all descendants for the CID comprises the steps of creating a local list of groups, creating a queue of groups Q, inserting CID and a contact for CID into Q including information identifying a descendant group  $G_N$ , queuing a reference to CID in L, dequeuing  $G_N$ , and when not a member of  $G_N$  contacting  $G_N$  and requesting a collection of all group attribute and contact information for  $G_N$ .

26. (Withdrawn) The method of claim 25, further comprising the steps of extracting child group  $G_C$  information from the group attribute and contact information for  $G_N$ , and when  $G_C$  is not in L adding  $G_C$  in L, queuing a reference to  $G_C$  in Q, and annotating a parent/child relationship in L for  $G_C$  and  $G_N$ , and when  $G_C$  is in L annotating a parent/child relationship in L for  $G_C$  and  $G_N$ .

27. (Cancelled)

28. (Cancelled)

29. (Withdrawn) A method of synchronizing information between an existing member and a new member after the new member has joined a peer-to-peer group, comprising the steps of calculating a signature of a database of the existing member, sending the signature to the new member wherein a comparison of information in the signature to the database of the new member is performed, receiving object requests for information from the existing member's database that are not present in the new member's database, and transmitting only the requested objects to the new member.

30. (Withdrawn) The method of claim 29, wherein the step of calculating a signature of the database comprises the steps of creating a sorted list of information from each object in the database, the information including a unique identifier UID, a sequence number SNUM, and an age.

31. (Withdrawn) A method of managing shared group information in a peer-to-peer group, comprising the steps of receiving a message having a unique identifier UID from a peer, searching a local database of shared group information for the UID, and when the UID does not exist in the local database determining that the message is new, updating the local database with the message, and flooding the message to neighbors.

32. (Withdrawn) The method of claim 31, when the UID exists in the local database further comprising the steps of comparing a sequence number of the message  $SNUM_M$  with a sequence number of the message in the local database  $SNUM_{dB}$ , and when the  $SNUM_{dB} > SNUM_M$  sending the message in the local database to the peer.

33. (Withdrawn) The method of claim 31, when the UID exists in the local database further comprising the steps of comparing a sequence number of the message  $SNUM_M$  with a sequence number of the message in the local database  $SNUM_{dB}$ , and when the  $SNUM_{dB}$  does not equal  $SNUM_M$  determining that the message is new, updating the local database with the message, and flooding the message to neighbors.

34. (Withdrawn) The method of claim 31, when the UID exists in the local database further comprising the steps of comparing a sequence number of the message  $SNUM_M$  with a sequence number of the message in the local database  $SNUM_{dB}$ , and when the  $SNUM_{dB}$  equals  $SNUM_M$  checking an age of the message  $AGE_M$  with an age of the message in the local database  $AGE_{dB}$ , and when the  $AGE_{dB} > AGE_M$  sending the message in the local database to the peer.

35. (Withdrawn) The method of claim 31, when the UID exists in the local database further comprising the steps of comparing a sequence number of the message  $SNUM_M$  with a sequence number of the message in the local database  $SNUM_{dB}$ , and when the  $SNUM_{dB}$  equals  $SNUM_M$  checking an age of the message  $AGE_M$  with an age of the message in the local database  $AGE_{dB}$ , and when the  $AGE_{dB} = AGE_M$  acknowledging receipt of the message from the peer.

36. (Withdrawn) The method of claim 31, when the UID exists in the local database further comprising the steps of comparing a sequence number of the message  $SNUM_M$  with a sequence number of the message in the local database  $SNUM_{dB}$ , and when the  $SNUM_{dB}$  equals  $SNUM_M$  checking an age of the message  $AGE_M$  with an age of the message in the local database  $AGE_{dB}$ , and when the  $AGE_{dB} < AGE_M$  determining that the message is new, updating the local database with the message, and flooding the message to neighbors.

37. (Withdrawn) A method of maintaining a peer-to-peer group graph, the group including a plurality of members each having a unique identifier, comprising the steps of calculating a group signature comprising a lowest of the unique identifiers of the members of the group, creating a group signature record containing the group signature, and flooding the group signature record to the members of the group.

38. (Withdrawn) The method of claim 37, wherein the step of calculating comprises the steps of comparing a received group signature record with a local unique identifier and when the group signature in the group signature record is larger than the local unique identifier, generating a new group signature record containing the local unique identifier as a new group signature, and flooding the new group signature record to the members of the group.

40. (Withdrawn) The method of claim 38, further comprising the step of waiting a predetermined time after the step of comparing before performing the step of generating the new group signature record.

41. (Withdrawn) The method of claim 40, wherein the step of waiting comprises the step of selecting a random time for the predetermined time.

42. (Withdrawn) The method of claim 40, wherein the step of waiting comprises the step of calculating a time based on a function of the size of the group and the likelihood that the local identifier should be the new group signature.

43. (Withdrawn) The method of claim 42, wherein the step of calculating a time based on a function of the size of the group and the likelihood that the local identifier should be the new group signature comprises the steps of estimating a size of the group  $N$  based on the current signature  $x$  as  $N = -\log(2)/\log(1-x)$  and estimating a probability that the local identifier should be the new signature as  $P(\text{other nodes} > \text{local ID}) = (1-ID)^{(N-1)}$ .

44. (Withdrawn) The method of claim 37, further comprising the steps of detecting a graph partition and reconnecting the graph.

45. (Withdrawn) The method of claim 44, wherein the step of detecting a graph partition comprises the steps of receiving a contact group record including a perceived group signature, and comparing the perceived group signature in the contact group record with the group signature record in the local database, and detecting a difference between the perceived group signature and the group signature record in the local database, and wherein the step of reconnecting the graph comprises the step of contacting the node that sent the contact group record.

46. (Withdrawn) A method of dynamically reconfiguring a peer-to-peer group graph to enhance performance thereof, comprising the steps of establishing a predetermined number of neighbor connections, assessing a utility of each of the neighbor connections, and disconnecting from a low utility neighbor when a number of neighbor connections exceeds the predetermined number.

47. (Withdrawn) The method of claim 46, wherein the step of establishing a predetermined number of neighbor connections comprises the steps of establishing a minimum number of neighbor connections, establishing a desired number of neighbor connections, and establishing a maximum number of neighbor connections, and wherein the step of disconnecting from a low utility neighbor when a number of neighbor connections exceeds the predetermined number comprises the step of disconnecting from a low utility neighbor when a number of neighbor connections exceeds the desired number of neighbor connections.

48. (Withdrawn) The method of claim 47, further comprising the step of refusing a connect request when the number of neighbor connections is equal to the maximum number of neighbor connections.

49. (Withdrawn) The method of claim 47, further comprising the step of connecting to a new neighbor when the number of neighbor connections is less than the minimum number of neighbor connections.

50. (Withdrawn) The method of claim 46, wherein the step of assessing the utility of each of the neighbor connections comprises the steps of establishing a utility index  $U(x)$  for each neighbor connection, upon receipt of a new object and upon receipt of an acknowledgment indicating a flooded object is new computing  $U(x) = (U(x) * (1 - 1/32)) + 128$  for that neighbor connection, and upon receipt of an old object and upon receipt of an acknowledgment indicating that a flooded object is old computing  $U(x) * (1 - 1/32)$  for that neighbor connection.

51. (Withdrawn) The method of claim 46, wherein the step of disconnecting comprises the step of sending a disconnect message containing contact information for at least one neighbor.

52. (Withdrawn) The method of claim 46, further comprising the steps of determining a diameter of the graph, and contacting a new neighbor when the diameter is greater than a predetermined diameter.

53. (Withdrawn) A method of departing a peer-to-peer group, comprising the steps of constructing a disconnect message including contact information for at least one neighbor and sending the disconnect message to all current neighbors.